**Explain how FreeNet enables individuals to publish anonymous web pages that are extremely difficult to censor. Does this mean that the battle against state censorship in oppressive regimes has been won?**

*Abstract*

*Anonymity is a necessity for free speech, without it democracy cannot exist. Ensuring that anonymity has been hard to do online, however Freenet is a novel attempt to ensure anonymity. By encrypting and fragmenting documents which are stored on multiple nodes, and ensuring that nodes can only talk to a small subset, rather than a whole network, an individual's anonymity becomes stronger. While Freenet does not make it impossible to trace what someone has downloaded, it does make it much harder. Freenet is a big step towards enabling people living in countries with oppressive governments to voice their views.*

**Why is Anonymity important?**

The first amendment in the United States constitution is arguably one of the most important, and most controversial articles of law. It states that individuals have the right to free speech, free from persecution. Democracy is based on individuals electing representatives, based on their performance and what they say they will do. If freedom of speech is denied, then it is not possible for anyone to critique the government, and evaluate whether they are truly carrying out their responsibilities, or whether another party would be more appropriate. Indeed, without free speech, it is

impossible for other parties to exist, as their views will inevitably contradict that of the current government in power. Free speech can be ensured by anonymity. If an oppressive government does not know who is criticizing it, then it cannot punish the individual.

An oppressive government could choose not only to punish the individual speaking (if they can identify them), but also those who have listened to them, as they may believe that this individual is right. However, anonymity also allows others to hear other views without fear.

**How can anonymity be achieved?**

When we seek anonymity in modern life, we have two broad options. One is to use a computer, and the other is to use non technological methods – for example, posting leaflets though a door. The latter, with the advent of CCTV, DNA profiling and fingerprint analysis, becomes very hard to do anonymously. If an individual (or a government) really want to find out who posted something, they now have that ability. While there are other non technological methods, they are beyond the scope of this essay.

There are varying degrees of technological anonymity. Firstly, one could adopt a pseudonym, or for forums simply be a “guest”. However this is merely not telling people your name, it does not prevent people from finding out who you are. There are numerous other ways of trying to protect your digital anonymity from things like IP Address traces, and other methods, but most, with enough time and effort can be cracked. Maybe a different approach is needed. Currently, we can find out that someone is looking at a document, and which document they are viewing. What if a system was set up whereby we know that “Mr Smith” is looking at something, but we don’t know what it is? This is the approach taken by Ian Clarke when he designed Freenet (Clar, 1999).

**The Freenet Approach**

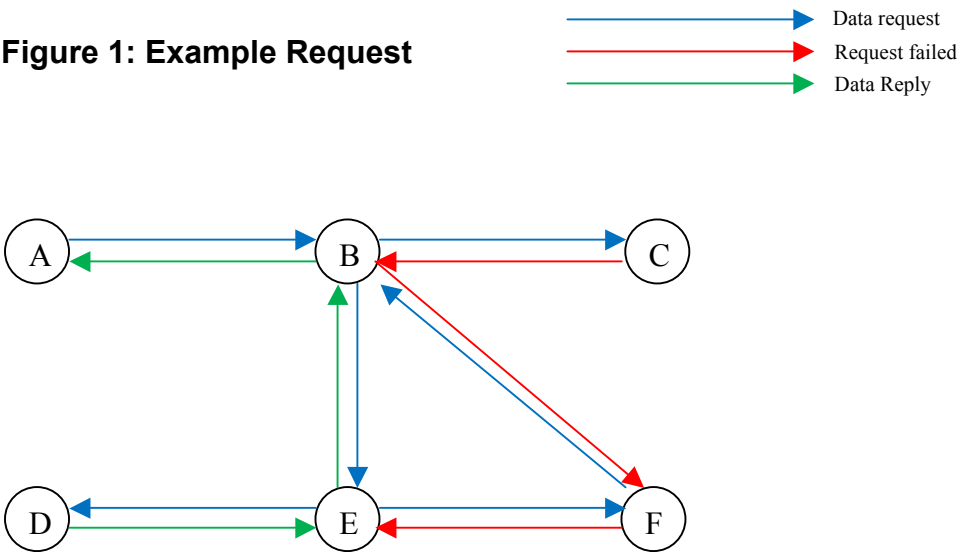
Users to Freenet sign up, and as payment, allow Freenet a portion of their hard drive and bandwidth to set up a node. This node is then integrated into the network by being assigned a random GUID (globally unique identifier), and creating a public and private key for itself. That node can then store encrypted information, and send it on request. Therefore, when someone uploads information to the network, it can be stored (multiple times) on the network and the creator does not have to be online. However, each node does not contain a full file. Rather than computers storing whole documents or webpage’s (which are easy to trace), Clarke’s algorithm breaks the file into fragments, all of which are then indexed, encoded and stored on multiple nodes (Clarke, 1999). This means that if a user was to look at what was stored on

the node on their computer, they would not be able to make sense of what was on it; even if they broke the encryption, they would only have document fragments.

File retrieval works in an innovative way. Sites like Gnutella search for a file by the first computer asking every computer it knows if they have the document, and then those computers ask all the computers they know for the document and so on. This is a very wasteful approach as there is no attempt to refine the search by determining the likelihood of which computers may have the file (Clarke, Miller, Hong, Sandberg, & Wiley, 2002). With Freenet, each node has links to some other nodes, and can also work out the likelihood that any given node it has links with will have a file. It then sends a request to the node it deems to be the most likely to return the right file. The node receiving the request is (at least initially) unlikely to have the file requested, so it then looks at its index of nodes, and forwards the request to the node it deems the next most likely. After a set number of “jumps” from one node to another, the request will terminate. However, hopefully before that happens, a node with the relevant file will be found, and it will then send its file back from node to node to the computer that originally requested it. The system also avoids loops and dead ends – should a node not have any links for a request, it sends back a message saying so, and the node that sent it the request will move to the second most likely. Equally, if a node realises that it has just received a request that it already forwarded on, it sends back a message saying there is a loop and that the sending node should try the node it ranks second most likely (Clarke, 2003)

This is shown in Figure 1

**Figure 1: Example Request**




An example request. A requests data from B. B forwards request to C. C has no relevant contacts and so sends a failure message to B. B forwards the message to E. E forwards to F. F forwards to B. B recognises a loop has been made and sends a loop message to F. F has no more relevant contacts and sends a failure message to E. E forwards to D. D has the data and forwards it to E. E forwards the data to B. B Forwards the data to A.

What helps ensure anonymity, is that when a node receives a request, it only knows which node it received the request from, and to which node it sent the request. For example, in the diagram above, node f is only aware of nodes e and b, it cannot tell that the request came from node a. Of course, as the original file is fragmented, several requests will have to be sent out for one node to locate the separate fragments of a file.

The end result is that while someone can tell that “Mr Smith” is downloading something, they do not know if it is for him, or if it is simply something that the node housed on his computer is downloading to pass on – in which case he has no way of viewing the data as it is encrypted and fragmented. If it is for Mr. Smith to see, no one can find out where it has come from originally, how to put the fragments he has downloaded together, or how to decrypt it.

**Problems with Freenet**

The main problem with Freenet is that it entails larger download times as documents are fragmented and spread over several nodes. Moreover, every node along the chain has to download and then pass along the file.

However, Freenet does try to combat this. It does this in two ways. Firstly, when a request is made, each node involved in the process takes note of whether the node it sent a message to had a link to the file. Therefore, if it was unsuccessful, the probability it assigns that node for being able to handle similar requests goes down, where as the probability it assigns for the node that passes back the file will greatly increase for that type of request (Clarke, et al., 2002). Therefore, in the diagram above, if node a made a similar request, node b may not forward it to node c, but to node e instead as node e had the file the last time it was requested.

The second way Freenet can speed up downloading is also done with nodes on the successful chain. When the file is returned though nodes a, b and e, nodes b and e may cache the file, so that if they get the same request, they can return the data, rather than forward on the request (Clarke, et al., 2002). This way, if node a made the same request, node b may forward to node c, but node c may have cached the file, and so, now the file only has to go though one node to get to the requester. This way, the files that are most requested become available on more nodes, and so become faster for everyone to download, while less popular ones take longer to download.

**Losing anonymity with Freenet**

Some have identified a fatal flaw with the Freenet system. As was pointed out earlier, nodes which regularly handle requests for the same file fragments will often store them themselves, thus shortening the download time. However, the node stored on the viewers computer is also part of that chain, and so, if repeatedly requested, the node on the computer will store that file. This means that without even being connected to the internet, you can “download” the file in seconds (Greene, 2005). Therefore, if a computer is seized, and computer analysts know what files you are likely to have viewed, they can see how long it takes to download a file. If it is seconds (i.e. downloading from the node on the computer) rather than hours (for a file stored on a remote node), then they know you have viewed that file. Therefore, if a dictatorial government doesn’t want someone viewing certain cites, and they suspect someone of viewing them. By taking their computer and trying to download these, they can tell which have been viewed.

**Has Freenet won the battle against state dictatorship and oppressive regimes?**

Clearly Freenet is not impenetrable. However, it seems unlikely that anything ever will be. For every advance computer science makes in ensuring anonymity or security, others will work equally as hard to find ways to crack or circumvent those advances. So it seems that true anonymity will never be achievable. But Freenet is still a big step in the right direction. It has taken a novel approach to ensuring anonymity, and seems likely to form the basis for further advances.

While individuals living in dictatorships or oppressive regimes cannot yet view sites totally without risk on freenet, the risk has been greatly reduced from what it was. While it seems likely that in time, forensic computer scientists working for such governments will find ways to track what people view, for the time being, it may have given thousands of people the chance to open their minds to new socio-political ideas and ideologies, although because of the nature of Freenet – we cant tell how many!

## References


Clarke, I. (1999). *A Distributed Decentralised Information Storage and Retrieval System.* Division of Informatics, University of Edinburgh.

Clarke, I. (2003, July 20). *Freenet's Next Generation routing Protocol.* Retrieved November 1, 2009, from Freenet: www.freenetproject.org

Clarke, I., Miller, S., Hong, T., Sandberg, O., & Wiley, B. (2002, January). Protecting free expression online with Freenet. *IEEE Internet Computing* , 40-49.

Greene, T. (2005, May 13). *I know what you downloaded from Freenet.* Retrieved November 1, 2009, from The Register: www.theregister.co.uk